

## **“ENSEÑANZA DE LA ALGORITMICA: En busca de una solución”**

**Lic. A. Rosso - Ing. M. Daniele**  
**arosso@exa.unrc.edu.ar**  
**mdaniele@exa.unrc.edu.ar**  
**Tel/fax: 058-676228**

Área de Computación  
Facultad Ciencias Exactas Físico-Químicas y Naturales  
Universidad Nacional de Río Cuarto  
Área Temática: Educación

### **Resumen.**

En la tarea de encontrar una mejor manera de enseñar las nociones básicas elementales de la algorítmica es que, desde el año 1993, comenzamos a trabajar en un proyecto pedagógico tendiente a elaborar una metodología, acorde con las tendencias actuales.

Con la implementación de la metodología, entre otras cosas, pudimos establecer y clasificar algunos de los errores que reiteradamente cometen los alumnos en el aprendizaje de ciertos temas.

Es objeto de este trabajo, la clasificación y distribución de los errores, como así también un análisis de las posibles causas de los mismos. Utilizamos como fuentes de información para elaborar la lista de los errores: los aportes de los docentes a cargo de las clases prácticas y teóricas, las falencias detectadas en las evaluaciones parciales y finales de los alumnos de Diagramación y Programación, asignatura de primer año de las carreras de Licenciatura y Profesorado en Ciencias de la Computación y Analista en Computación de la Facultad de Ciencias Exactas Físico-Químicas y Naturales de la U. N. R. C.

Hemos elaborado una clasificación de los errores y analizado sus posibles causas. A la vez que podemos realizar una presentación según su tipo y según su frecuencia de ocurrencia, en los años 1994, 1995 y 1996.

## 1. Introducción

El tiempo, la experiencia -propia y ajena, los resultados obtenidos, las discusiones con colegas, las respuestas de los alumnos, las dificultades que se presentan a la hora de resolver un problemas planteando una solución algorítmica, nos convencieron de la impostergable necesidad de encontrar la forma adecuada para inducir al uso de técnicas y metodologías que aporten mejoras muy significativas a la correctitud de los planteos de soluciones para un problema dado y disminuyan, con importante notoriedad, la ocurrencia de errores sistemáticos.

En el año 1993, generamos un proyecto pedagógico innovador, para ser aplicado a la asignatura Diagramación y Programación que se dicta en el primer año de las carreras de Ciencias de la Computación. En dicho proyecto se sugiere la enseñanza de la programación tendiente a:

- Poner de relieve todos los factores que intervienen en la solución del problema, lo que implica entender el problema en todos sus detalles, para poder determinar los lineamientos principales que le darán solución.
- Inducir a utilizar técnicas que permitan la resolución de problemas planteados. Lo que implica conocer y saber aplicar distintas formas para plantear y resolver situaciones.
- Emplear esquemas correctos de tratamiento de secuencias, los que establecen una técnica en el uso de las estructuras de control.
- Simbolizar los procesos de solución en un lenguaje particular, parecido al lenguaje natural, que no es libre, sino que está sujeto a ciertas reglas de sintaxis y con una semántica. A este lenguaje lo llamamos “pseudo\_código”. Este pseudo\_código estructurado, es capaz de adaptarse a las especificaciones del problema, permitiendo escribir un algoritmo de fácil lectura a la vez que estructura la solución del problema y es independiente del lenguaje de programación.
- Generar un algoritmo, expresado en el pseudo\_código, que establezca la íntima relación entre datos y las estructuras de control que permiten su tratamiento.

Como corolario de estos pasos, el algoritmo es traducido a un lenguaje de programación estructurado y es procesado en una computadora.

Este proyecto pedagógico fue sufriendo modificaciones en pos de mejorar la enseñanza y el aprendizaje de la programación, del cual surgieron otros proyectos pedagógicos todos con el mismo objetivo: “*lograr que los alumnos sean capaces de resolver problemas, planteando soluciones informatizables, a través de programas que sean correctos y relevantes.* Ello, mediante el estudio de los algoritmos y de las estructuras de datos, a fin de determinar la representación más adecuada a la solución del problema, independiente de un lenguaje de programación particular”.

Con la implementación de la metodología propuesta en los proyectos pedagógicos mencionados, hoy, nos encontramos abocados a analizar críticamente si estamos dando respuesta a alguno de los problemas que hemos detectado.

Para ello realizamos un análisis de la modificación en la distribución de los errores que hemos clasificado. Parámetro que nos indicará en cuanto los ajustes implementados en los proyectos pedagógicos están siendo adecuados.

## 2. Planteo del Problema

Para tener un mayor contacto con los alumnos, hemos modificado la forma de organizar el dictado de las clases, implementando clases teórico-prácticas, a partir del año 1996, con la finalidad de trabajar con grupos más pequeños de alumnos y poder realizar un seguimiento de la evolución de los grupos y sus aprendizajes. Con el claro objetivo de lograr que los alumnos sean capaces de resolver problemas, planteando soluciones informatizables.

Creemos que el planteo del problema supone un reto para el estudiante, movilizándolo a encontrar una respuesta. Cuando elabora la solución correcta reafianza su técnica de resolución. Cuando comete errores debe construir nuevamente el concepto obteniendo una nueva solución, que de cuenta del problema, pero quedando de manifiesto la razón de sus errores.

Compartiendo las ideas de Dijkstra y Hoare, que sostienen que muchos errores de programación pueden evitarse haciendo a los programadores conscientes de métodos y técnicas que ellos aplican instintivamente. Para que la metodología empleada tenga resultados positivos es necesario conocer los errores más frecuentes y sus causas. Así podemos prever algunos errores, atacando las razones que los motivan.

## 3. Resultados

Para la construcción de algoritmos correctos es necesario la observación de diferentes pasos, la no observación de alguno de estos puntos con el debido cuidado, hace que se cometan errores en el planteo de la solución. Centrándonos ahora en los errores que aparecen en las producciones de los alumnos podemos inferir que los mismos tienen relación con el no cumplimiento de, por lo menos, alguno de los puntos señalados.

### a) En cuanto a los errores detectados

- I. Mal manejo del código de programación
- II. Mal manejo de variables.
- III. Mal manejo de las estructuras
  - a) de programación
  - b) de datos
- IV. Errores propios del problema a resolver

Las fuentes de información que utilizamos para elaborar esta lista de errores posibles, son:

1. El análisis de una muestra de 300 evaluaciones escritas que realizan los alumnos (desde 1994 a 1996), para cumplir con las exigencias de la asignatura Diagramación y Programación, que se dicta en el primer año de las carreras de ciencias de la computación de nuestra universidad.
2. Los aportes que realizan los docentes de la asignatura, según lo observado en las clases de aula y en las consultas individuales.
3. Las falencias detectadas cuando se evalúa a los alumnos en el examen final para la aprobación de la asignatura.
4. Las consultas de los alumnos.

### b) En cuanto a las causas de los errores cometidos

Las razones o causas de la ocurrencia de los errores que se desarrollan a continuación son inferencias que realizamos desde nuestra interpretación teórica del problema. Su verificación requerirá un trabajo posterior.

ERRORES	Tipo	CAUSA
- Confunde el uso de los operadores Y (and) y O (or), de > (mayor) y < (menor)	I	No puede manejar los operadores lógicos, ni los operadores relacionales. No puede discriminar entre condición de unión y disyunción.
- No puede expresar la solución con la sintaxis usada, falla la notación algorítmica especialmente en la expresión de las condiciones	I	La falta de interiorización con el pseudo código utilizado provoca errores de sintaxis. Esto podría deberse a una falta de dedicación por parte del alumno a estudiar paulatinamente los contenidos impartidos en las clases teóricas.

ERRORES	Tipo	CAUSA
- Mezcla los tipos de variables, hace asignaciones y comparaciones inválidas. - Define variables como parámetros de una función y en el cuerpo de la misma usa otras.	II	Cuando uno define una variable debe definir de que tipo es; con ello se caracteriza el conjunto de valores al que pertenece. El tipo de un dato no puede ser deducido del contexto, cuando se programa, entonces debe declararse explícitamente. Los alumnos que no tienen claro lo expuesto en el párrafo anterior, mezclan los tipos en asignaciones o comparaciones ya que no pueden detectar claramente el conjunto de valores de los cuales las variables toman sus valores (iniciales o finales).

ERRORES	Tipo	CAUSA
- Confunde, mezcla los conceptos de función y acción. En casos como: a) define una función que retorna más de un valor. b) define una acción que retorna un solo valor y luego la invoca como función.	IIIa	El alumno tiene problemas con la definición formal de función y su diferencia con las acciones. No tiene una clara conceptualización matemática de función. La dificultad para establecer la diferencia entre parámetros formales y parámetros reales se produce, al no poder establecer la diferencia entre caso particular y caso general.
- Lee en el cuerpo de una función o acción las variables que pasa como parámetros.		
- Define una función que no devuelve ningún valor.		
- Invoca una función o acción (con pasaje de parámetros por valor) sin valores de entrada para los parámetros o valores para los cuales la misma no está		

definida.		
- Ciclos sin fin, no avanza nunca dentro del ciclo.	IIIa	El error se comete porque el alumno no usa o no respeta, los esquemas de recorrido de secuencias, en ocasiones porque no los interpreta y en otras porque no se convence de su uso. El sujeto no puede modelar su problema, ajustándolo a un modelo correcto y probado ya existente.
- Avanza en un ciclo sin considerar la condición que indica la marca de fin de la estructura utilizada		
- No inicializa variables utilizadas como contadores o acumuladores.		
- No controla cuando la secuencia está vacía al comenzar el ciclo.		
- No inicializa la/s variables que usa para presentar una condición, o plantea mal la condición. No inicializa la adquisición de elementos de la secuencia a tratar.		
- Mezcla estructuras como SEGÚN <var> HACER con SI <cond> ENTONCES SINO	IIIa	Las causas de éstos errores es que el alumno no alcanza a identificar los casos posibles para determinar la utilización de la estructura correcta. Si para una condición dada los casos posibles son dos, si la unión de estos casos dan el todo, deberá utilizar la composición alternativa ( Si <condición> entonces <consecuencia1> sino <consecuencia2>). Si independientemente de los casos posibles sólo interesa un caso, deberá utilizar la composición condicional (Si <condición> entonces <consecuencia1>). Mientras que, si se requiere dar una respuesta a todos los casos posibles deberá usar la composición selectiva (Según <var> hacer ). Si el alumno no puede determinar todos los casos posibles para la solución a un problema dado, tampoco podrá realizar un buen planteo de las condiciones.
Actualiza o lee la variable índice utilizada para un ciclo PARA.	IIIa	Desconoce la sintaxis de la sentencia. No entiende o no puede diferenciar las distintas estructuras repetitivas, la forma en que se toma el siguiente elemento de una secuencia en cada una de ellas.

ERRORES	Tipo	CAUSA
- Mal manejo de los índices	IIIb	Cuando se trabaja con arreglos se maneja una

cuando se utiliza arreglos.		estructura que identifica sus elementos con un índice.
- Confunde filas con columnas.	IIIb	Sobre el conjunto de los índices está definida una relación de orden, que induce un orden en la ubicación de los elementos. Los errores en el manejo de estos valores puede ocasionar dos problemas: 1. Que se busquen elementos fuera del rango permitido. 2. Mala manipulación de la información. Estos errores muestran que el alumno no incorpora el concepto, no puede establecer la relación entre el índice y el orden inducido sobre la ubicación de los datos.
- Mal manejo de las operaciones sobre pilas y colas cuando el soporte es una lista.	IIIb	No logra abstraerse a la definición de pila o cola, independizándose del soporte utilizado. Problema que no evidencia cuando el soporte es un arreglo.
- Mal enganche de los datos en listas encadenadas.	IIIb	La manipulación de datos en memoria dinámica es relativamente simple cuando se maneja con gráficos, pero no es tan sencilla cuando se lo codifica. Requiere un proceso de abstracción donde se independice del lugar físico que ocupa el dato y establezca su conexión lógica con el resto de la secuencia. La razón de cometer este error es no tener claramente identificada la técnica a utilizar para los movimientos habituales de las estructuras que los utilizan (inserción, supresión, listado, etc). No tienen internalizado el mecanismo a utilizar; aunque gráficamente pueden plantearlo, la abstracción al código no se puede realizar, quedándose solo en la etapa de lo gráfico.

ERRORES	Tipo	CAUSA
- No muestra los resultados obtenidos. - No resuelve el problema planteado.	IV	Hay errores que provienen de una interpretación incorrecta de las consignas, quizás debido a una lectura apresurada o superficial, ello no le permite realizar el planteo del problema a resolver. Resuelve uno similar, en el mejor de los casos.
- Faltan casos de tratar. - No considera secuencias unitarias. - No ve la manipulación de la cabeza de una lista como un caso particular.	IV	El sujeto no puede realizar todas las combinaciones posibles abstrayéndose de los objetos mismos, trata de resolver el problema preocupado por los detalles de la sintaxis y no analiza los casos particulares atinentes al problema.
- Resuelve el problema para un determinado número de pasos, no puede generalizar.	IV	En algunos casos no puede abstraerse de ejemplos particulares. No es capaz de realizar una generalización o caracterización global.

## b) En cuanto al análisis de la información relevada

Se tomó una muestra de 300 exámenes parciales de los años 1994, 1995 y 1996 pertenecientes a los alumnos que cursaron la asignatura Diagramación y Programación (asignatura anual que se dicta en el primer año de las carreras de Ciencias de la Computación). Se analizaron 100 parciales de cada año.

Se analizó cada examen parcial, clasificando y cuantificando los errores cometidos según la clasificación en las cuatro categorías de errores antes descriptas. Se obtuvo un total de 314 errores, considerando que algunos alumnos no cometen ningún error y otros más de uno. Los datos cuantificados están volcados en las siguientes tablas por categoría y por año relevado:

### I. Mal manejo del código de programación

AÑOS	1994	1995	1996	TOTAL
CANTIDAD DE ERRORES	1	1	0	2

### II. Mal manejo de variables

AÑOS	1994	1995	1996	TOTAL
CANTIDAD DE ERRORES	23	20	10	53

### III. Mal manejo de las estructuras

#### a) de programación

AÑOS	1994	1995	1996	TOTAL
CANTIDAD DE ERRORES	51	39	41	131

#### b) de datos

AÑOS	1994	1995	1996	TOTAL
CANTIDAD DE ERRORES	13	4	13	30

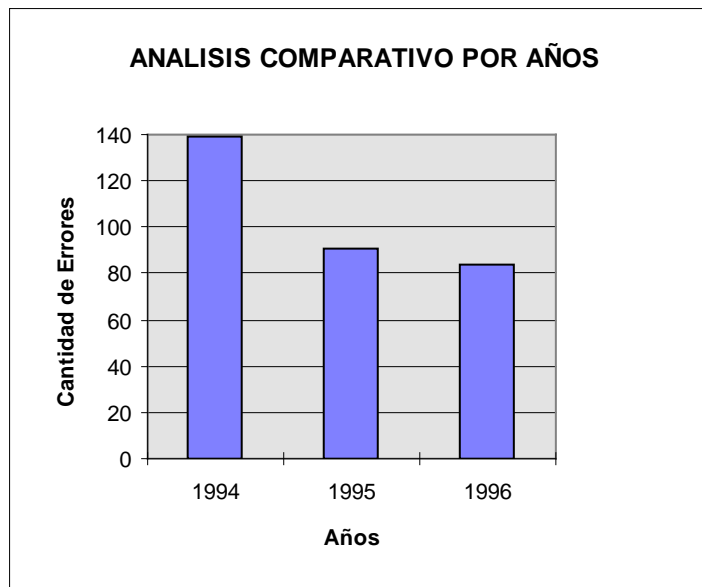
### IV. Errores propios del problema a resolver

AÑOS	1994	1995	1996	TOTAL
CANTIDAD DE ERRORES	51	27	20	98

TOTAL GENERAL DE ERRORES      139 + 91 + 84      = 314

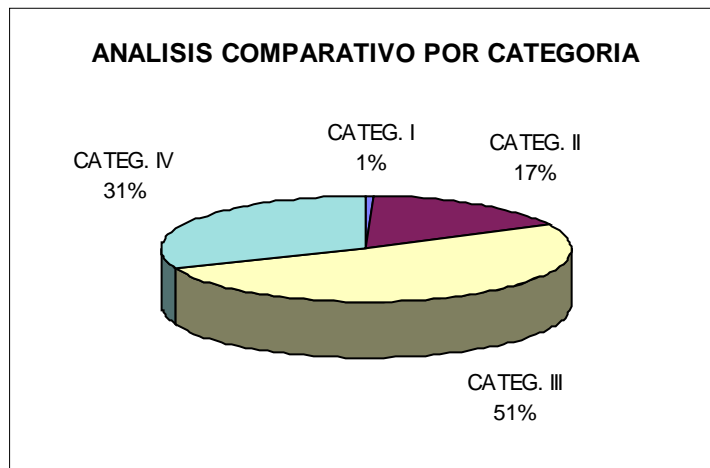
Con estos resultados, se hace un análisis gráfico comparativo en dos direcciones; por un lado se trata de comparar según los años y las metodologías de trabajo implementada en cada una de ellos, y por el otro, se compara la ocurrencia de los errores según cada categoría.

En el siguiente gráfico, se ve reflejado la totalidad de errores encontrados para cada uno de los años relevados. Se observa que hay una notable diferencia a favor, en los años 1995 y 1996. Esto no revela que las mejoras impuestas a la metodología de trabajo que se sigue desde 1993, están realizadas en la dirección correcta.. Además, aunque en menor escala, vemos que entre el año 1995 y el año 1996 sigue disminuyendo la ocurrencia de errores sistemáticos. Estos resultados nos dicen que la metodología propuesta en el proyecto pedagógico está tomando la firmeza buscada para el control de los errores detectados.

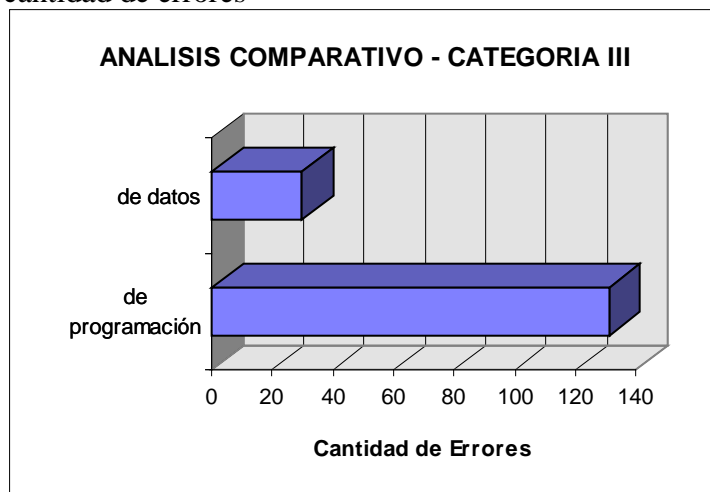


El gráfico que sigue muestra los totales de errores para los tres años, diferenciados por categorías. Podemos observar que la cantidad de errores que caen en la categoría I son no significativos y es de esperar dicho resultado. Los que caen en categoría II, si bien son el 17 % del total, en el relevamiento de los datos se observó que la mayor ocurrencia de estos se da en los primeros exámenes y no así en los últimos, entonces podemos decir que el alumno va internalizando paulatinamente los conocimientos, con el avance en el cursado de la asignatura va capitalizando sus experiencias y evita algunos de estos errores. A pesar de ello, esto no nos libera de afianzar en las actividades prácticas propuestas que tiendan a mejorar la ocurrencia de estos errores, hasta que desaparezcan.





El siguiente gráfico hace un análisis más detallado del grupo donde se cometen la mayor cantidad de errores



### **CONCLUSIONES FINALES**

Desde el año 1993 se desarrolla en la asignatura Diagramación y Programación de las carreras de Ciencias de la Computación de nuestra Universidad, un proyecto pedagógico que contempla un abordaje de la enseñanza de los contenidos superadora de otra modalidad que consideramos no suficientemente adecuada para la enseñanza de los temas.

En el marco del desarrollo de dicho proyecto aparecieron errores sistemáticos en las producciones de los alumnos, que analizamos e intentamos una clasificación como así también cual fue su comportamiento a lo largo de los años que llevamos trabajando con los proyectos y los ajustes que hemos realizado a lo largo de sus reiteradas implementaciones.

En base al análisis realizado, podemos decir que hemos mejorado en algunos aspectos tales como la ocurrencia de ciertos errores, y su distribución a lo largo del dictado de la asignatura. también es claro que otros todavía persisten y es necesario hacer un estudio más profundo a fin de establecer posibles estrategias de enseñanza para que la ocurrencia de errores disminuya.

## Bibliografía

- [1]- A ROSSO, M. DANIELE, G. ROJO, G. FRASCHETTI. Proyecto Pedagógico Innovador para la enseñanza de la Algorítmica. Fac. Cs. Extas. Fco-Qcas y Naturales. Dpto. Matemática U.N.R.C. 1993.
- [2] A. ROSSO, J. GUAZZONE. Errores que se cometen al resolver un problema con estructuras de repetición. Dpto. de Matemática. Fac. Cs. Extas. Fco-Qcas y Naturales. U.N.R.C. 1995.
- [3] A ROSSO, M. DANIELE. Algunos errores sistemáticos detectados en el proceso de aprendizaje de la Algorítmica. 4to. Ateneo de Profesores Universitarios de Computación y Sistemas. San Luis Noviembre de 1996.
- [4]- P.C. SCHOLL - J.P. PEYRIN. Esquemas Algorítmicos Fundamentales. Secuencias e Iteración. Ed. Masson, 1991.
- [5]- J.D. NOVAK - D.B. GOWIN. Learning how to learn. New York. Cambridge University Press, 1984.
- [6]- MARK ALLEN WEISS. Estructuras de datos y algoritmos. Ed. Addison-Wesley Iberoamericana, 1995.
- [7]- J. GLENN BROOKSHEAR. Introducción a las Ciencias de la Computación. Ed. Addison-Wesley Iberoamericana, 1995.
- [8]- NIKLAUS WIRTH. Algoritmos + Estructuras de datos = Programas. Ed. del Castillo S.A., 1980.
- [9]- CAIRÓ / GUARDATI. Estructuras de Datos. Ed. McGraw-Hill Interamericana de México, 1993.